Using 2D and 3D Convolutional Neural Networks to Predict Semen Quality

Jon-Magnus Rosenblad[1], Steven Hicks[2, 3], Håkon Kvale Stensland[1],
Trine B. Haugen[3], Pål Halvorsen[2, 3], Michael Riegler[2, 4]
[1]Simula, Norway, [2]SimulaMet, Norway, [3]Oslo Metropolitan University, Norway,
[4]Kristiania University College, Norway

ABSTRACT

In this paper, we present the approach of team *Jmag* to solve this year's Medico Multimedia Task as part of the MediaEval 2019 Benchmark. This year, the task focuses on automatically determining quality characteristics of human sperm through the analysis of microscopic videos of human semen and associated patient data. Our approach is based on deep convolutional neural networks (CNNs) of varying sizes and dimensions. Here, we aim to analyze both the spatial and temporal information present in the videos. The results show that the method holds promise for predicting the motility of sperm, but predicting morphology appears to be more difficult.

1 INTRODUCTION

In an effort to explore how medical multimedia can be used to create performant and efficient prediction algorithms, the 2019 Multimedia for Medicine Task [6] focuses on the analysis of microscopic videos of human semen to predict certain quality characteristics of spermatozoon. The challenge presents three different tasks, of which we decided to focus on the tasks which are required in order to participate this years challenge, namely, the *prediction of motility task* and the *prediction of morphology task*. Motility and morphology are two metrics which are commonly used to determine the quality of a semen sample. Motility is the analysis of how each spermatozoon moves and is primarily split into three different categories; progressive, non-progressive and immotile. Morphology refers to the shape and size of the sperm and may be split into three groups; sperm with head defects, tail defects, and midpiece defects. More information about the dataset can be found in the original publication [5].

2 APPROACH

Motility and morphology are properties of sperm which appear differently in the videos human semen. Motility may be difficult to assess looking only at the spatial dimension, as it is heavily dependent on the temporal information present in a video. By contrast, morphology is highly dependent on the visual features of the sperm and not necessarily their movement, although there may be some correlation between the the movement and the morphology, i.e., a sperm with a tail defect may move slower. Consequently, predicting these two aspects of semen require different approaches. To preserve the temporal and the spatial information in a video when predicting motility, we use 3D convolutional neural networks (CNNs). When predicting morphology, we discard the temporal information and make a prediction based on a single frame using a 2D CNN. For the morphology approach, we use a higher resolution on the video when predicting morphology to preserve the minor details present in the sperms appearance. In the following two sections, we present our approach of using CNNs to solve the requires sub-tasks of this year's medico task.

Motility

We present two methods for predicting the motility. First, we use a simple 3D CNN to see how well a model using just a few layers performs on this task. Second, we present a deeper and more complex 3D CNN to see how this improves over the simpler model. The simple model uses a very shallow network architecture consisting of only two convolutional layers. Each convolutional layer extracts 32 filters using a kernel size of $4 \times 4 \times 4$ and $5 \times 5 \times 5$ respectively, which the output is then passed to a fully-connected layer before making the prediction. The complex model consists of three consecutive convolutional blocks, where each block is made up of three convolutional layers and a pooling layer to reduce the spatial and temporal dimensions. Following the conventions of Li et al. [8], we add a $1 \times 1 \times 1$ convolutional layer at the end of the block to act as a pixel-wise fully-connected layer over the filters. The architecture for the complex and simple model can be found in Figure 1c.

Due of the limited amount of data, we perform data augmentation during training. First, we extract 20 random samples from a single data point for which we perform several augmentation techniques including random crops, noise injection, and vertical/horizontal flips. To decrease training time, we first downsample the resolution of each sample to 128×171 pixels for both the training and the validation dataset, then we randomly select samples of consecutive 15 frame intervals and randomly crop the image to 128×128 pixels. The frame samples are then randomly flipped both horizontally and vertically with a probability of 0.5 each. Finally, we add some noise injecting each pixels with some random values selected from a uniform distribution in the interval $[-0.01, 0.01]$. For validation, we split each video into blocks consisting of 15 consecutive frames and discard the frames that remain. Each frame block is then cropped into a 128×128 at the upper left edge of the frame. We then calculate the average prediction score over all blocks of a video for each video, and take the average of these averages to get our final prediction score. We do this to avoid weighing longer videos more than shorter ones in our final score and rather weigh each video the same.

Morphology

To predict morphology, we use a relatively deep 2D CNN while avoiding making it deep in order to avoid vanishing gradients [2, 4]. The network consists of 5 convolutional layers, each with kernel size 4×4 and strides 4×4 and 1×1 alternating starting with 4×4. They pad with zeros to keep it's initial size before striding. They

Copyright 2019 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).
MediaEval'19, 27-29 October 2019, Sophia Antipolis, France

CEUR-WS.org/Vol-2670/MediaEval_19_paper_63.pdf

Method	Fold	Prog		Non-Prog		Immotile		Mean	
		MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
Simple	1	11.05	13.38	7.70	10.46	14.26	19.55	11.00	14.95
	2	9.66	12.87	6.87	7.91	26.75	31.64	14.43	20.24
	3	40.93	44.53	8.44	10.67	11.81	16.10	20.39	28.02
	Mean	20.55	23.59	7.67	9.68	17.61	17.61	15.27	21.07
Complex	1	9.34	11.54	8.51	10.29	10.38	14.13	9.41	12.09
	2	10.08	12.72	5.71	6.88	7.76	10.71	7.85	10.39
	3	11.05	13.35	8.01	10.18	8.62	11.33	9.23	11.69
	Mean	10.16	12.54	7.41	9.12	8.92	12.06	8.83	11.39
ZeroR	1	18.01	21.05	8.03	9.91	15.59	22.47	13.88	18.68
	2	18.88	22.06	7.62	8.61	14.27	17.44	13.59	16.98
	3	15.45	17.74	9.46	11.61	11.37	14.04	12.09	14.68
	Mean	17.45	20.37	8.37	10.12	13.74	18.31	13.19	16.86

Table 1: The results for the prediction of motility task.

Method	Fold	Head		Midpiece		Tail		Mean	
		MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
Single Frame	1	2.40	2.74	8.36	9.44	8.68	11.20	6.48	8.60
	2	2.73	3.17	8.19	9.87	6.93	9.95	5.92	8.29
	3	2.88	3.40	8.45	10.59	6.55	8.63	5.96	8.13
	Mean	2.67	3.10	8.33	9.97	7.39	9.93	6.13	8.38
ZeroR	1	1.90	2.36	8.73	9.86	7.33	9.33	5.99	7.95
	2	2.72	3.17	8.01	9.76	7.25	9.99	5.99	8.27
	3	2.22	2.98	8.47	10.70	6.76	8.66	5.82	8.13
	Mean	2.28	2.86	8.40	10.12	7.11	9.34	5.93	8.12

Table 2: The results for the prediction of morphology task.

have 32, 128, 128, 512, and 512 filters each respectively. After the final convolutional layer, we pass the output through three fully-connected layers; one with 1024 nodes, one with 512 nodes and the output layer with 3 nodes. Each layer in the network uses the activation function ReLU, except for the output layer which uses a linear activation.

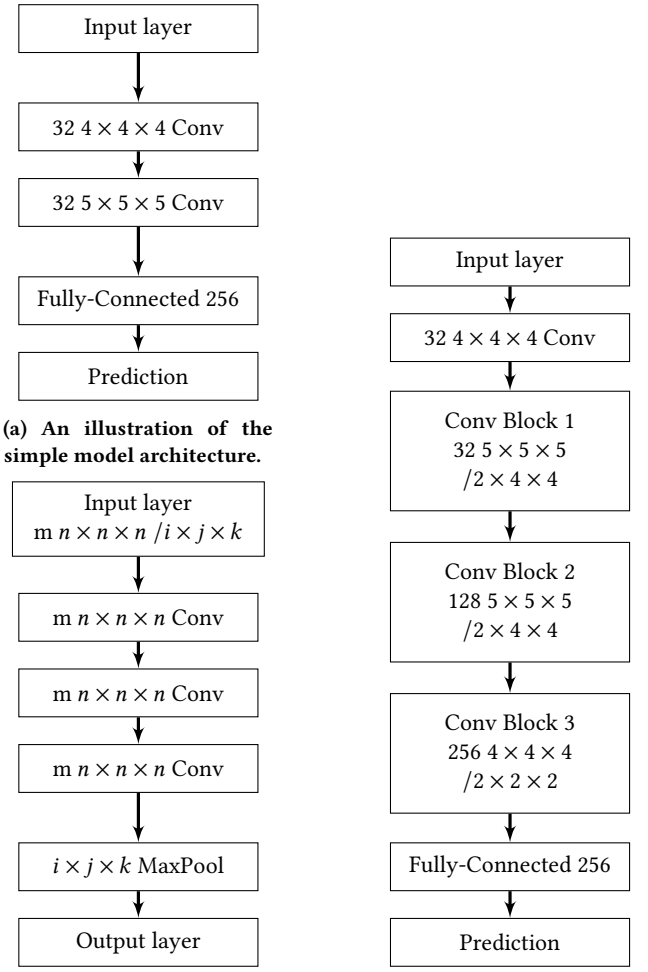
For both training and validation, data was prepared similarly to that of the motility experiments, the only difference being that we used a single frame to make predictions at a resolution of 240×320 and did not perform any cropping. We still, however, performed noise injection with noise retrieved from the same distribution, and random flips with using the same probabilities.

Training

All models were trained for a maximum of 200 epochs, only interrupting the training if the evaluation loss did not improve over the last 10 epochs. The models were trained using the deep learning library Keras [3] with a TensorFlow [1] back-end. The experiments were run on a machine consisting of a single Nvidia RTX 2080Ti graphics card, 128 GB of RAM, and an Intel Xeon Gold 5120 CPU clocked at 2.20 GHz. Each motility model was trained with a batch size of 64 using the Adam optimizer [7] configured as described in the original paper. The morphology model was trained using the same configuration, only with a smaller batch size of 16.

3 RESULTS AND DISCUSSION

Looking at the motility experiments (Table 1), we see that the complex model achieves much better results than the simple model. It is clear that our complex model is able to extract more crucial information from the data to make better predictions. Comparing the complex model to the ZeroR baseline, we see a mean absolute error (MAE) improvement of 0.0436 which shows that the deep learning at the very least is able to learn to associate some movement of the sperms with the related motility values. For the morphology experiments (Table 2), we see that our model fails to beat predicting the mean value of the labels (ZeroR). It fails to learn the individual shape of each sperm and collectively predict total of each category. For future work, we will increase the size of the network to make it more adaptable, which may bring other challenges such as making the network harder to train due to the increased risk of vanishing gradients [2, 4].



(a) An illustration of the simple model architecture.

(b) An illustration of the convolutional block used in the complex model.

(c) An illustration of the compelx model architecture.

Figure 1: The CNN architectures used for the prediction of motility task.

4 CONCLUSION

In this paper, we presented the work done as part of the Medico Multimedia Task where we participated in two of the three available subtasks. We used deep CNNs for both tasks, where we achieved an average MAE of 0.0883 for the motility task and an average MAE of 0.0613 for the morphology task.

REFERENCES

[1] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. 2015. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. (2015). https://www.tensorflow.org/ Software available from tensorflow.org.

[2] Yoshua Bengio, Patrice Simard, Paolo Frasconi, and others. 1994. Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks* 5, 2 (1994), 157–166.

[3] François Chollet and others. 2015. Keras. https://keras.io. (2015).

[4] Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*. 249–256.

[5] Trine B. Haugen, Steven A. Hicks, Jorunn M. Andersen, Oliwia Witczak, Hugo L. Hammer, Rune Borgli, Pål Halvorsen, and Michael A. Riegler. 2019. VISEM: A Multimodal Video Dataset of Human Spermatozoa. In *Proceedings of the 10th ACM on Multimedia Systems Conference (MMSys'19)*. https://doi.org/10.1145/3304109.3325814

[6] Steven Hicks, Pål Halvorsen, Trine B Haugen, Jorunn M Andersen, Oliwia Witczak, Konstantin Pogorelov, Hugo L Hammer, Duc-Tien Dang-Nguyen, Mathias Lux, and Michael Riegler. 2019. Medico Multimedia Task at MediaEval 2019. In *CEUR Workshop Proceedings - Multimedia Benchmark Workshop (MediaEval)*.

[7] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[8] Min Lin, Qiang Chen, and Shuicheng Yan. 2013. Network in network. *arXiv preprint arXiv:1312.4400* (2013).